

日本語・満族語の辞書作成のための 補助システム (VI)

本 田 道 夫

- I. はじめに
- II. 新コード体系
- III. その他の変更・改良
- IV. おわりに

I. はじめに

日本語・満族語の辞書作成のための補助システムの開発に関連して、最終的な印刷を綺麗にするために、満族語文字について印刷用フォントの改良を考えていたが、辞書作成者の依頼により、約15万字ほどの非常に多くのフォントを作成し無料で提供してきた文字鏡研究会で作成していただけることになった。それに伴い、無駄な文字の作成をお願いすることを避けることもあり、これまでの利用経験から満族語文字体系の見直し・整理が行われ、かなり大幅な文字の削除と追加が行われた。

文字コード体系はコンピュータで処理する場合には最も基本的な部分の一つであり、追加はあっても、削除はないと思っていたことではあるが、今回のものは整理された文字体系であり、今後は大きな変更はないものと考えてその変更に対応して、文字コード体系の再整理を行った。また、満族語文字体系の文字コードの変更に伴って、文字鏡文字の文字コードについても変更することとした。

そして、これまで開発してきた各種のデータファイルやプログラムファイルについては、新文字コード体系に対応した変更が必要であり、その作業も行った。

なお、これまでの辞書データの入力・編集に利用しているシステムについての利用経験から要望のあったことも含めて、少し改良を行った。

II. 新コード体系

II.1 文字コードの配置

これまでの満族語文字体系は、半角文字 110 個、全角文字 62 個、ローマ字表現用の半角文字の大きさの特殊英文字 5 個であったが、変更は、半角文字について 17 個を削除し 2 個を追加、全角文字は 3 個削除であり、新文字体系では、半角文字 95 個、全角文字 59 個、特殊英文字 5 個である。もちろん、削除されたところの文字コードは空きのままとして利用せず、新規文字については、その空いたところあるいは文字コード表の最後に追加するという対応もあったが、それではコード表が虫食い状態となること、また満族語文字の途中に特殊英文字が入ることから、この際、虫食い状態がなく、満族語文字半角文字の後に英特殊文字を配置するようにコードの再割り当てを行った。満族語全角文字は半角文字の後に配置した。

最初に満族語・日本語の辞書作成のための補助システムについて作成依頼のお話をいただいたときは、上記のように、満族語文字は、半角 110 文字、全角 62 文字程度、通常のシフト JIS には含まれないが利用したい漢字（これに対しては文字鏡文字を利用することにした）が約 2,000 個ということであったので、文字鏡用漢字には、1 バイト目に 16 進数表現で F0（以後、16 進数表現を 0x をつけて 0xF0 のように表現することにする）から 0xF9 を、2 バイト目には 0x21 から 0xFE を用いることにし、満族語文字の 1 バイト目には、全角文字は 0xFA、半角文字は 0xFB を用いることとしていた。しかし、実際に日本語・満族語辞書の作成を進めていく途中で、文字鏡文字の利用個数が徐々に増えて

- 1) 実際の満族語文字は、大まかには半角程度の大きさのものと全角程度の大きさのものの 2 種類の文字に分けられるが、日本語文字のように半角・全角と正確には区別できず、半角・全角と分類している文字も、それぞれ、多少大きさは異なる。しかし、入力・編集システムの開発を簡単にするために、パソコンでの画面表示用は、大きさを英文字と同じ大きさの半角文字、漢字と同じ大きさの全角文字としている。ただし、最終的な辞書印刷は、本当の文字の大きさを正確に反映できるものとしている。

2,000個を超え、さらにその後もまだまだ増える可能性もあったので、途中から、2バイト目にはこれまで利用していなかった16進数で0x00から0x20の部分も用いるようにして対応してきた。しかし、あまり増えると文字鏡文字(1バイト目0xF0~0xF9)の後に満族語文字(1バイト目0xFAと0xFB)、さらに、そのあとに文字鏡文字(1バイト目0xFC~0xFE)ということにもなりかねないので、この際、文字鏡文字についてもコードを変更することとし、満族語文字、および文字鏡文字のコード配置は次の表1のようなものとした。な

表1 新文字コード体系の満族語文字、特殊文字、文字鏡文字のコード配置

(以下のコード表示は、16進数表示である)

満族語文字半角	F021 ~ F06F
特殊英文字(半角)	F070 ~ F074
満族語文字全角	F081 ~ F0BA
文字鏡文字	F100 ~ F9FE (現在までの利用)

表2 満州語文字および特殊英文字の新文字コード表

満州文字半角 (normalsize)

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
F020		ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ
F030	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ
F040	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ
F050	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ
F060	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ
F070	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ

満州文字全角 (normalsize)

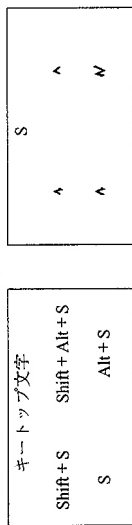
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
F080		ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ
F090	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ
F0A0	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ
F0B0	ᠠ	ᠡ	ᠢ	ᠣ	ᠤ	ᠥ	ᠦ	ᠨ	ᠪ	ᠬ	ᠭ	ᠱ	ᠳ	ᠷ	ᠸ	ᠠ

図 1 キーボードでの満州語文字および特殊英文字の入力配置

○満州語文字・キーボード対応モード (Ctrl-3) でのキーボードの文字配置

半角/ 全角	1 ハ	2 カ	3 キ	4 ク	5 ケ	6 コ	7 カ	8 キ	9 ク	0 ケ	1 コ	2 カ	3 キ	4 ク	5 ケ	6 コ	BS
Tab	Q ツ	W フ	E ト	R ニ	T ハ	Y シ	U ス	I セ	O ソ	P テ	Enter						
Caps Lock	A ア	S イ	D ウ	F エ	G オ	H カ	J キ	K ク	L ケ	Enter							
Shift	Z ザ	X ジ	C ジ	V セ	B ソ	N タ	M チ	Enter	Shift								

○Shift キー、およびAlt キーとの併用の見方 (「S」のキーを例に)
 ・次図の「キートップ文字」は、Shift キーも、Alt キーも押さずに押下したときに入力される文字を示している。



○満州語文字・キーボード対応モード (Ctrl-3) でのキーボードの文字配置

半角/ 全角	1 ハ	2 カ	3 キ	4 ク	5 ケ	6 コ	7 カ	8 キ	9 ク	0 ケ	1 コ	2 カ	3 キ	4 ク	5 ケ	6 コ	BS
Tab	Q ツ	W フ	E ト	R ニ	T ハ	Y シ	U ス	I セ	O ソ	P テ	Enter						
Caps Lock	A ア	S イ	D ウ	F エ	G オ	H カ	J キ	K ク	L ケ	Enter							
Shift	Z ザ	X ジ	C ジ	V セ	B ソ	N タ	M チ	Enter	Shift								

お、文字鏡文字のコードは1バイト目が0xF0は0xF1に、0xF1は0xF2というように1増加させたが2バイト目については変更はしていない。

満州語文字および特殊英文字の新文字コード表を表2に示す。

キーボードでの満族語文字および特殊英文字の入力配置を図1に示す。

II.2 ファイルの変換あるいは作成

文字コード体系の変更に伴って、次のようなファイルの変換、変更あるいは作成が必要であった。

- (A) フォントファイル (変更) : 削除文字のところに後方から文字の移動と追加
 - ・満族語文字用のフォントファイル (ManRF.ttf, ManRP.ttf の2ファイル)

- (B) 文字コード表など (新規作成)
 - ・満族語文字コード表
 - ・満族語文字入力キーボード配置
 - ・文字鏡文字コード表

- (C) データファイル (変換)
 - ・MD 000.DBD ~ MD 024.DBD (25 ファイル)
 - ・MD 旧清語.xml
 - ・12 本文.xml
 - ・清文總彙全巻.xml

- (D) プログラム関連のファイル (変更)
 - (a) 辞書データ入力・編集システムのプログラム (1プログラム)
 - (b) 印刷システムのプログラム (1プログラム)
 - (c) ローマ字表現の英文字列を満族語文字列に変換するための規則表
 - (d) 上の(c)規則表をプログラムに取り入れるような形式変換のプログラム

(5 ファイル)

(C)のデータファイル中の外字文字について：

旧の満族語全角の文字コード領域としていたのは1バイト目が16進数表現で0xFAであるが、2バイト目は0x80から0xBDまでしか利用しておらず、全てを使用していたわけではなかった。そして、今回の変換に際して、辞書データとして作成されてきていたファイルには、1バイト目が0xFAで、2バイト目が旧文字では使用していない0xBE以降の文字コードが含まれていたものがあった。調べてみると、16進数で0xFA40～0xFC4Bまでの外字領域の文字のうち、たまたま満族語文字に用いてなかったコードの文字が利用されていた。外字については、最初のシステム設計の段階で、漢字コードの配置を調べたときに、外字領域については、その領域の記述が0xF040～0xF9FF、あるいは0xF000～0xFFFCなどと書かれたものなどがあった。そのいずれの記述でもユーザが自由にそこに文字を作成できる領域となっていたので、どれが正しいかはそれ以上は調べずに、その一部を満族語文字と文字鏡文字用に用いることにし、特に特定の文字が割り当てられているとは思っていなかった。今回のことがあり、改めて、調べてみると、Windows XPでは0xFA40～0xFC4Bにはあらかじめ文字(外字)が割り当てられているようであった。そして、たまたま辞書データ入力・編集システムで用いていなかったコードの文字を利用していただようである。その外字文字コードは新文字コード体系でも用いないので、現在はそのままにしている。もし、文字鏡文字が増えた場合には、その文字を文字鏡文字に置き換えるなどの対応が必要となるが、そのときに対応することにした。

(D)のプログラム関連のファイルについて：

(c)の「ローマ字表現の英文字列を満族語文字列に変換するための規則表」の各行は、

ローマ字列	満族語文字列	満族語文字列の16進数表現	コメント
-------	--------	---------------	------

となっており、第3項目の「満族語文字列の16進数表現」は通常の英数字での表現である。そのために、他のデータファイルに適用した変換のプログラムは、満族語文字列の部分は変換するが、通常の英数字の部分は変換しないので、変換した満州文字列に対する16進数表現の英文字列で置き換えるということが必要であり、このファイルだけは、別の変換プログラムを作成する必要があった。

Ⅲ. その他の変更・改良

Ⅲ.1 満族語文字のローマ字表現用特殊英文字の入力

満族語のローマ字表現のときに、5文字ほど通常の英文字には含まれていない文字(特殊英文字)が必要であるので作成している。これまでは、満族語データ入力・編集システムで、その特殊英文字を入力するには、メニューバーの「文字入力」項目から、「特殊英文字モード」を選択して入力するようにしていた。その入力を終了して、通常英文字を入力するには、同様に、メニューバーの「文字入力」項目から、「通常文字モード」を選択する必要があった。特殊英文字の入力は通常の英文字入力中にごく稀に行うものであり、連続して入力することはほとんどあり得ないとのことで、いちいち入力モードを切り替えるのは面倒であるとのことで、通常文字入力の時にAltキーを併用することにより入力できるようにした。これに伴ってメニューバーからの「特殊英文字モード」は無くした。

Ⅲ.2 Altキー併用時の処理の変更

これまで、辞書データ入力・編集システムで、特殊英文字などのAltキー併用でのキー入力はWin 32 APIでのキーボード入力の処理でWM_MENUCHARで捕捉し処理していたが、この方法では警告音のような音が出ていた。いろいろと調べていたが、WM_SYSCHARで捕捉し処理すると音がしなくなることが分かり、この方法に変更した。Windows上でのプログラムを開発した当初からの懸案事項であり、時々インターネットで検索したり、何か対処方法を探していたりしたときに試したりしていたが、なかなか解決しなかった。分かっ

てみれば簡単なことであったが、長年の懸案が一つ解決した。

IV. おわりに

文字コードはシステム全体の基盤的なものであり、それを変更することは、2個のフォントファイルの変更、ほとんどすべてのデータファイルとプログラムファイルの変更の必要があった。これだけに全ての時間を費やしたわけではないが、プログラムの詳細を忘れていたので、プログラムの処理方法を思い出しながらの作業であり、変更を完了したつもりでも一部変更を忘れていたこともあり、プログラムファイルの変更とデータファイル変換用のプログラムの作成にはほとんど2ヶ月程かかった。

もちろん、その期間内のデータ入力・修正は、これまでの文字コード体系で行ってもらい、プログラムの変更が終了し、その後にデータ変換のプログラムが完成したあと、データ入力・修正を中断して、全てのデータファイルを送ってもらい一気に変換して、それを送り返すと同時に、新フォントファイル、文字コード表、キーボード入力表、編集システムと印刷システムのプログラムなどを一緒に送った。変換プログラムが完成しているのだから、プログラム適用による変換はほとんど時間がかからず、受け取ってから送り返すまでは、1時間程度であった。

いずれにしても、全体の基盤とも言うべきところを変更することは想定していなかったために、関連するものが何か、どこに変更すればよいかなどの洗い出しも含めて結構な作業であった。